Referência Rápida Debian

Osamu Aoki <osamu@debian.org>
Paulo Rogério Ormenese (líder: pt-br) <pormenese@uol.com.br>
'Autores' on page 27

1.08-3, Dom Mai 1 07:45:18 UTC 2005

Resumo

Esta Referência Rápida Debian (http://qref.sourceforge.net/) dedica-se a fornecer uma breve introdução ao sistema Debian como uma **referência rápida**. Isso é um excerto do Referência Debian (http://qref.sourceforge.net/).

Nota de Copyright

Copyright © 2001–2005 por Osamu Aoki <osamu@debian.org>.

Este documento pode ser usado sob os termos da GNU General Public License versão 2 ou superior. (http://www.gnu.org/copyleft/gpl.html)

É concedida a permissão para fazer e distribuir cópias literais deste documento contanto que o aviso de direito de cópia (copyright) e este aviso de permissão sejam preservados em todas as cópias.

É garantida a permissão para copiar e distribuir versões modificadas deste documento sob as condições da cópia literal, contanto que o trabalho derivado resultante seja totalmente distribuído sob os termos de um aviso de permissão idêntico a este.

É garantida a permissão para copiar e distribuir traduções deste documento em outro idioma sob as condições acima para versões modificadas, exceto que este aviso de permissão pode ser incluído em traduções aprovadas pela Free Software Foundation em vez de no inglês original.

Sumário

1 Prefácio **1**

1.1 Convenções do documento 1

1.2 Básico das distribuições Debian 1

2 Atualizando uma distribuição para a `testing` **3**

2.1 Transição do APT para a versão Woody 3

2.2 Preparação para a transição (`stable` para `testing`) 3

2.3 Atualizar o sistema Debian 4

2.3.1 Melhor prática de atualização usando o `dselect` 5

3 Gerenciamento de pacotes Debian **7**

3.1 Introdução 7

3.1.1 Ferramentas principais 7

3.1.2 Ferramentas úteis 8

3.2 Básico do gerenciamento de pacotes Debian 8

3.2.1 Instalar *tarefa* com o `tasksel` ou `aptitude` 8

3.2.2 Configure o sistema APT 9

3.2.3 `dselect` 9

3.2.4 `aptitude` 10

3.2.5 Comandos `apt-cache` e `apt-get` 10

3.2.6 Seguindo um tipo da distribuição Debian 12

3.2.7 Rebaixando todos os pacotes para a `estável` 12

3.2.8 Visão geral do arquivo `/etc/apt/preferences` 12

3.3 Comandos de sobrevivência do Debian 13

3.3.1 Verificar bugs no Debian e procurar ajuda . . . 14
3.3.2 Resolvendo problemas com atualização pelo APT . . . 14
3.3.3 Resgate usando o `dpkg` . . . 15
3.3.4 Recuperação dos dados de seleção de pacotes . . . 16
3.3.5 Resgate do sistema após quebrar `/var` . . . 16
3.3.6 Instalar um pacote em um sistema não iniciável . . . 16
3.3.7 O que fazer se o comando `dpkg` estiver quebrado . . . 17
3.4 Comandos de nirvana do Debian . . . 17
3.4.1 Informação sobre um arquivo . . . 17
3.4.2 Informação sobre um pacote . . . 18
3.4.3 Instalação com APT sem acompanhamento . . . 19
3.4.4 Reconfigurar pacotes instalados . . . 19
3.4.5 Remover e eliminar (purge) pacotes . . . 20
3.4.6 Mantendo pacotes antigos . . . 20
3.4.7 Sistema híbrido `stable/testing/unstable` . . . 20
3.4.8 Descartar arquivos de pacote em cache . . . 21
3.4.9 Gravar/copiar a configuração do sistema . . . 21
3.4.10 Portar um pacote para o sistema `stable` . . . 21
3.4.11 Repositório local de pacotes . . . 22
3.4.12 Converter ou instalar um pacote binário alien . . . 23
3.4.13 Verificar arquivos de pacotes instalados . . . 23
3.5 Outras peculiaridades do Debian . . . 24
3.5.1 O comando `dpkg-divert` . . . 24
3.5.2 O pacote `equivs` . . . 24
3.5.3 Comandos alternativos . . . 24
3.5.4 `init` System-V e níveis de execução (runlevels) . . . 25
3.5.5 Serviços daemon desabilitados . . . 25

A Apêndice **27**
A.1 Autores . . . 27
A.2 Garantias . . . 29
A.3 Retorno . . . 29

Capítulo 1

Prefácio

Matenha-o pequeno e simples (KISS) é meu princípio guia.

1.1 Convenções do documento

Este documento "Referência Rápida Debian" fornece informações através de pequenos comandos do shell Bash.

Referência para:
- uma **página de manual Unix** é indicada na forma `bash(1)`.
- uma **página GNU TEXINFO** é indicado na forma `info libc`.

1.2 Básico das distribuições Debian

O Debian é oferecido em 3 "sabores" de distribuições :

- `stable`: Boa para usar em um servidor de produção. Chato para uma estação de trabalho (WS).
- `testing`: Boa para usar em uma WS.
- `unstable`: Nunca use-a cegamente.

Leia pelo menos a lista de discussão chave `debian-devel-anounce@lists.debian.org` para atualizações sobre o estado do Debian.

Em Março de 2002, estas três versões de distribuições correspondiam às versões de codinomes Potato (qualidade de produção), Woody (beta-teste, já bastante estável quando o lançamento era iminente), e Sid (alfa-teste). Em Agosto de 2002, logo após o lançamento do Woody, as distribuições correspondiam a Woody (qualidade de produção), Sarge (beta-teste, teria algumas

arestas ruins por um tempo) e Sid (sempre alfa-teste). Quando os pacotes na `unstable` não possuem bugs críticos ao lançamento (RC) cadastrados para eles após a primeira semana ou mais, eles são automaticamente promovidos para a `testing`.

Há duas maneiras de habilitá-lo a executar as últimas versões de software em um sistema.

- 'Atualizando uma distribuição para a `testing`' on the next page (principalmente para própositos de WS)
- 'Portar um pacote para o sistema `stable`' on page 21 (principalmente para prósitos de servidor)

Capítulo 2

Atualizando uma distribuição para a `testing`

2.1 Transição do APT para a versão Woody

Atualização pela rede do sistema APT e de alguns pacotes principais para a versão Woody pode ser feita como a seguir depois de incluir as fontes `stable` no arquivo `/etc/apt/sources.list` se você ainda usar o Potato.

```
# apt-get update
# apt-get install libc6 perl libdb2 debconf
# apt-get install apt apt-utils dselect dpkg
```

2.2 Preparação para a transição (`stable` para `testing`)

Seguir o sabor `testing` do Debian tem o efeito colateral de fornecer atualizações de segurança muito lentamente. Então fique alerta.

Atualização pela rede para `testing` pode ser feita como explicado a seguir (execute o script go-woody (http://www.debian.org/doc/manuals/debian-reference/examples/) para fazer isso em um comando):

Esvazie o arquivo `sources.list` existente:

```
# cd /etc/apt
# cp -f sources.list sources.old
# :>sources.list
```

Obtenha uma lista limpa de repositórios, para `stable`:

```
# cd /
# apt-setup noprobe
 ... selecione repositórios, acessados com métodos HTTP ou FTP
```

Adicione a seção `testing` a essa nova lista. As linhas `deb-src` são comentadas.

```
# cd /etc/apt
# grep -e "^deb " sources.list  >sources.deb
# grep -e "^deb-" sources.list  >sources.src
# sed -e "s/stable/testing/"  sources.deb \
     >>sources.list
# sed -e "s/stable/testing/" sources.src | \
     sed -e "s/^deb-/#deb-/"  >>sources.list

# apt-get update
# apt-get install apt apt-utils
# cat >preferences <<EOF
 Package: *
 Pin: release a=testing
 Pin-Priority: 600

 Package: *
 Pin: release a=unstable
 Pin-Priority: 50

 EOF
```

Opcionalmente, adicione a seção "unstable" dos repositórios.

```
# sed -e "s/stable/unstable/" sources.deb \
     >>sources.list
# sed -e "s/stable/unstable/" sources.src | \
     sed -e "s/^deb-/#deb-/"  >>sources.list
```

Veja 'Básico do gerenciamento de pacotes Debian' on page 8 para conhecer a arte do ajuste do `/etc/apt/sources.list` e do `/etc/apt/preferences`.

Agora você pode atualizar a lista de pacotes e atualizar pacotes, usando métodos descritos na próxima seção.

2.3 Atualizar o sistema Debian

Depois de configurar os arquivos `/etc/apt/sources.list` e `/etc/apt/preferences` apropriadamente, o sistema pode ser atualizado para `testing`. Consulte 'Gerenciamento de pacotes Debian' on page 7 para saber o básico, e veja 'Resolvendo problemas com atualização pelo APT' on page 14 se encontrar problemas.

2.3.1 Melhor prática de atualização usando o dselect

Se um sistema tem muitos pacotes que incluem pacotes -dev, etc., recomenda-se o método a seguir usando o dselect para um controle de pacotes mais refinado.

```
# dselect update  # sempre faça isso antes de atualizar
# dselect select  # selecione pacotes adicionais
```

Todos os seus pacotes atuais serão selecionados quando o dselect iniciar. O dselect pode oferecer-lhe a opção de instalar pacotes adicionais, baseado em depends, suggests, e recommends. Se você não quiser adicionar nenhum pacote, simplesmente pressione Q para sair do dselect novamente.

```
# dselect install
```

Você precisará responder algumas questões de configuração de pacotes durante essa parte do processo, então mantenha suas notas prontas e reserve algum tempo a essa parte. Consulte 'dselect' on page 9.

Use o dselect. **Ele sempre funciona :)** Se você precisar atualizar sem o dselect depois do Woody, considere usar o aptitude e outras opções.

Capítulo 3

Gerenciamento de pacotes Debian

Para reduzir o tráfego de rede nos repositórios Debian, certifique-se de configurar um proxy HTTP local usando o `squid` para os pacotes os quais o download tenha sido feito através do APT e, caso necessário, define a variável de ambiente `http_proxy` ou defina o valor `http` no arquivo `/etc/apt/sources.list`. Isso melhora consideravelmente a performance de atualizações em rede, especialmente em diversas máquinas Debian em uma LAN (rede local).

Apesar do recurso de pinagem do `apt_preferences(5)` ser poderoso, ele não resolve todos os problemas de dependências uma vez que os requerimentos das dependências tendem a puxar versões mais novas de pacotes de programas fundamentais.

+ O uso do método chroot é desejável para assegurar simultaneamente tanto a estabilidade do sistema quanto o acesso às últimas versões de softwares.

Este capítulo é baseado em um sistema Woody mas a maioria da informação também se aplica a um sistema Potato (exceto para o `apt_perferences(5)` e tópicos relacionados ao arquivo `/etc/apt/preferences`).

3.1 Introdução

Se ler toda a documentação para desenvolvedor é muito para você, leia este capítulo primeiro e comece a desfrutar de toda a capacidade do Debian com `testing/unstable` :-)

3.1.1 Ferramentas principais

```
dselect   - ferramenta de gerenciamento de pacotes orientada a menu (nível to
dpkg      - instala pacote (centrado em arquivo-pacote)
apt-get   - instala pacote (centrado em repositório de pacotes, CLI APT)
tasksel   - instala tarefa (um conjunto de pacotes)
aptitude  - instala pacote (pacote & tarefa, APT ncurses)
deity     - APT ncurses alternativo
synaptic, gsynaptic - alternativas GUI APT
```

Essas não são ferramentas de mesmo nível. O `dselect` roda no topo do APT (os comandos de linha são `apt-get`) e `dpkg`.

O APT usa `/var/lib/apt/lists/*` para rastrear pacotes disponíveis enquanto que o `dpkg` usa `/var/lib/dpkg/available`. Se você instalou pacotes diretamente usando `apt-get` ou programas similares como o `aptitude`, certifique-se de atualizar o arquivo `/var/lib/dpkg/available` a partir do item de menu `[U]pdate` no `dselect` ou a partir do comando shell "`dselect update`" antes de executar `dselect select`, `tasksel`, ou `dpkg -l`.

Como dependências de pacotes, o `apt-get` automaticamente insere pacotes com **depends** mas deixa os pacotes com **recommends** e **suggests**, enquanto o `dselect` oferece controle guiado por menus sob as opções desses pacotes e lhe pergunta pela escolha de pacotes baseado em **depends**, **recommends**, e **suggests**. O `aptitude` oferece a opção de inserir todos os pacotes automaticamente baseado em **depends**, **recommends** e **suggests**.

3.1.2 Ferramentas úteis

```
apt-cache         - verifica o repositório de pacotes no cache local
dpkg-reconfigure  - reconfigura um pacote já instalado
                    (se ele usar debconf)
dpkg-source       - gerencia arquivo de pacote fonte
dpkg-buildpackage - automatiza a construção de um arquivo de pacote.
...
```

3.2 Básico do gerenciamento de pacotes Debian

Você pode instalar um conjunto de pacotes chamados *tarefa* ou instalar pacotes individuais ou atualizar o sistema usando ferramentas de gerenciamento de pacotes como as ferramentas citadas abaxio.

3.2.1 Instalar *tarefa* com o `tasksel` ou `aptitude`

O `tasksel` é o **Instalador de Tarefas Debian**, que é oferecido como a opção "`simple`" durante o sistema de instalação.

Quando se precisa instalar uma função comum que requer vários pacotes, essa é a melhor maneira de fazê-lo. Execute os comandos como a seguir:

```
# dselect update
# tasksel
```

O `aptitude` também oferece acesso a tarefas seletivamente através de menus. Ele não somente lhe permite selecionar *tarefas* mas também lhe permite desmarcar para instalação pacotes que fazem parte de uma *tarefa* seletivamente atráves de um menu.

3.2.2 Configure o sistema APT

Se você tentar seguir um ambiente misto como descrito aqui, pode comumente chegar a alguns conflitos de dependências de pacotes. É uma boa idéia não misturar sabores. O que vem a seguir é para pessoas que desejam experimentar reconhecendo alguns riscos.

Para atualização seletiva seguindo a distribuição `testing`, o sistema APT (>Woody) precisa ser configurado como em 'Transição do APT para a versão Woody' on page 3 para usar o recurso `apt_preferences(5)`.

Primeiro, adicione os fontes para `stable`, `tyesting`, e `unstable` em seu arquivo `/etc/apt/sources.list`. Edite então o arquivo `/etc/apt/preferences` para definir o Pin-Priority apropriado. [1]

```
Package: *
Pin: release a=stable
Pin-Priority: 500

Package: *
Pin: release a=testing
Pin-Priority: 600

Package: *
Pin: release a=unstable
Pin-Priority: 50
```

3.2.3 `dselect`

Quando iniciado, o `dselect` automaticamente seleciona todos os pacotes "Requeridos", "Importantes" e "Padrão". No sistema Potato, alguns programas grandes como o teTeX e o Emacs se encaixavam nessas categorias e era melhor não instalá-los na instalação inicial manualmente desmarcando para instalação os mesmos (digitando '_'). No Woody, esses pacotes foram movidos para a categoria de pacotes "Opcional".

O `dselect` possui uma interface com o usuário de certa forma estranha. Existem quatro comandos ambíguos (Maiúsculas significa MAIÚSCULAS") :

```
Tecla-Chave   Ação
Q             Finalizar. Confirma a seleção atual e finaliza de qualquer form
              (sobrescreve dependências)
R             Reverter ! Eu não queria dizer isso.
D             Droga ! Não me importa com o que o dselect acha. Faça isso !
U             Marque todos com o estado sUgerido
```

[1] Eu mantive a descrição por aqui como está para manter a consistência com outras partes. Se você realmente está seguindo a `testing` ou `unstable`, você deve na verdade remover referências à `stable` em `/etc/apt/sources.list` e `/etc/apt/preferences`. Isso é porque a `testing` começa como uma cópia da `stable`.

Com `D` e `Q`, você pode selecionar seleções conflitantes a seu próprio risco. Lide com esses comandos com cuidado. Atualmente, o `dselect` é a ferrementa guiada por menus madura que oferece um controle refinado sobre as escolhas de **suggests** e **recommends**.

Adicione uma linha contendo a opção "expert" no arquivo `/etc/dpkg/dselect.cfg` para reduzir o barulho. Para uma máquina mais lenta, você pode executar o `dselect` em uma outra máquina mais rápida para encontrar os pacotes e pode usar `apt-get install` para instalá-los.

O `dselect` não fornece acesso a pacotes que não sejam os preferidos pelo Pin-Priority.

3.2.4 `aptitude`

O `aptitude` é um novo instalador de pacotes guiado por menus similar ao `dselect`. Ele pode também ser usado como um comando alternativo de linha de comando no lugar do `apt-get`. Consulte a página de manual do `aptitude(1)`.

O `aptitude` aceita comandos de uma única tecla, geralmente em letras minúsculas.

```
Tecla-Chave     Ação
F10             Menu
?               Ajuda para as teclas de comandos
u               Atualiza informações dos pacotes do repositório
g               Faz o download e instala os pacotes selecionados
q               Finaliza a tela atual e salva as mudanças
x               Finaliza a tela atual e descarta as mudanças
Enter           Visualiza informações sobre um pacote
```

O `aptitude` oferece a opção de fazer o download de todos os pacotes automaticamente baseado nos **depends**, **recommends**, e **suggests**. Você pode mudar esse comportamente escolhendo `F10 -> Opções -> Gerenciamento de Dependências` nos menus do aptitude.

O `aptitude` oferece acesso a todas as versões de um pacote.

3.2.5 Comandos `apt-cache` e `apt-get`

Seguindo a distribuição `testing` como descrito no exemplo acima, podemos gerenciar o sistema usando os seguintes comandos:

- `apt-get -u upgrade`

  Esse comando segue a distribuição `testing` e atualiza todos os pacotes no sistema enquanto instala suas dependências da distribuição `testing`;

- `apt-get -u dist-upgrade`

  Esse comando segue a distribuição `testing` e atualiza todos os pacotes do sistema enquanto instala e resolve suas dependências da distribuição `testing`.

- `apt-get -u dselect-upgrade`

  Esse comando segue a distribuição `testing` e atualiza todos os pacotes no sistema pela seleção do `dselect`.

- `apt-get -u install` *`pacote`*

  Esse comando instala o pacote *pacote* e suas depedências da distribuição `testing`.

- `apt-get -u install` *`pacote`*`/unstable`

  Esse comando instala o pacote *pacote* da distribuição `unstable` enquanto instala suas dependências da distribuição `testing`.

- `apt-get -u install -t unstable` *`pacote`*

  Esse comando instala o pacote *pacote* da distribuição `unstable` enquanto instala suas depedências também da distribuição `unstable` definindo o Pin-Priority de `unstable` para 990.

- `apt-cache policy` *`foo bar ...`*

  Esse comando checa o estado dos pacotes *foo bar . . . .*

- `apt-cache show` *`foo bar ...`* `| less`

  Esse comando checa as informações dos pacotes *foo bar . . . .*

- `apt-get install` *`foo=2.2.4-1`*

  Esse comando instala a versão específica *2.2.4-1* do pacote *foo*.

- `apt-get -u install` *`foo bar-`*

  Esse comando instala o pacote *foo* e remove o pacote *bar*.

- `apt-get remove` *`bar`*

  Esse comando remove o pacote *bar* mas mantém intactos os arquivos de configuração que tenham sido modificados/personalizados.

- `apt-get remove --purge` *`bar`*

  Esse comando remove totalmente o pacote *bar* com todos os arquivos de configuração modificados/personalizados.

Nos exemplos acima, o uso da opção `-u` acima assegura que o `apt-get` exibirá uma lista de todos os pacotes que serão atualizados antes de efetuar a ação. O trecho a a seguir define a opção `-u` como padrão:

```
$ cat >> /etc/apt/apt.conf << .
//Sempre exibe pacotes a serem atualizados (-u)
APT::Get::Show-Upgraded "true";
.
```

Use a opção `-s` para simular um atualização sem na verdade fazê-la.

3.2.6 Seguindo um tipo da distribuição Debian

Dependendo de suas preferências por um sabor do Debian a seguir, você pode mudar o exemplo de `/etc/apt/preferences` em 'Configure o sistema APT' on page 9 para se adequar a suas necessidades :

```
seguir a estável:           mude o Pin-Priority da testing para 50
seguir a testing:           mantenha as confgiurações como exibidas acima
seguir a testing(unstable): mude o Pin-Priority da unstable para 50
seguir a unstable(testing): mude o Pin-Priority da unstable para 800
```

Uma linha guia para a escolha do Pin-Priority é mover de cima para baixo na tabela acima conforme o tempo passa de um momento imediatamente após o lançamento de uma distribução para um momento do congelamento (freeze) para o próximo lançamento.

Atenção: seguir o tipo `testing` do Debian possui o efeito colateral de obter atualizações de segurança muito lentamente.

Caso você misture os tipos do Debian `testing` com `stable` ou `unstable` com `stable`, você provavelmente irá obter também muitos pacotes-chave não intencionalmente do `testing` ou `unstable` que poderão conter bugs. Portanto, esteja avisado sobre esse problema.

Exemplos do arquivo `/etc/apt/preferences` que prendem alguns pacotes chave para suas versões mais maduras enquanto seguem as versões menos maduras de outros pacotes não-essenciais estão disponíveis no subdiretórios de exemplos (`http://www.debian.org/doc/manuals/debian-reference/examples/`) com o nome de `preferences.testing` e `preferences.unstable`. Por outro lado, o arquivo `preferences.stable` força todos os pacotes a serem rebaixados para suas versões existentes na distribuição "estável" (um "downgrade" na verdade).

3.2.7 Rebaixando todos os pacotes para a `estável`

Para rebaixar todos os pacotes para `stable`, edite `/etc/apt/preferences` como a seguir:

```
Package: *
Pin: release a=stable
Pin-Priority: 1001
```

e execute "`apt-get upgrade`", o que força o rebaixamento devido ao Pin-priority > 1000. Fique atento, podem haver alguns pequenos problemas com dependências.

3.2.8 Visão geral do arquivo `/etc/apt/preferences`

No arquivo `/etc/apt/preferences` de exemplo apresentado em 'Configure o sistema APT' on page 9, "Package: *" na primeira linha significa que esta seção se aplica a todos os pacotes. Você pode usar um nome de pacote ao invés de "*" para especificar o Pin-Priority desse pacote.

Na segunda linha, ”Pin: release a=stable“ significa que o `apt-get` obtém o arquivo `Packages.gz` de um repositório contendo o arquivo `Relese` que contenha a linha ”Archive: stable“.

Na última linha ”Pin-Priority: 500“ especifica o Pin-Priority como 500.

Caso existam diversos pacotes com o mesmo nome, normalmente aquele com o valor de Pin-Priority mais alto é o que será instalado.

Os significados de Pin-Priority são :

- 1001 e superior: Prioridades de ”downgrade“ (rebaixamento de versões).

  O ”downgrade“ (rebaixamento de versão) do pacote é permitido para esta faixa de Pin-Priority.

- 100 a 1000: Prioridades ”Standard“ (Padrão).

  O ”downgrade“ do pacote não é permitido. Alguns valores chave de Pin-Priority :

  - 990: a prioridade definida pela opção `--target-release>` ou `-t` do `apt-get(8)`.
  - 500: a prioridade para todos os arquivos de pacotes padrão.
  - 100: a prioridade para os arquivos de pacotes atualmente instalados.

- 0 a 99: Prioridades não-automáticas. (Estas são usadas somente caso o pacote não esteja instalado e não exista nenhuma outra versão disponível.)

- menor que 0: A versão não é selecionada nunca.

Os mesmos efeitos de `--target-release` podem ser obtidos configurando o arquivo `/etc/apt/apt.conf` como abaixo :

```
# echo 'APT::Default-Release "testing";' >> /etc/apt/apt.conf
```

A opção de linha de comando `--target-release` e a configuração no arquivo `/etc/apt/apt.conf` têm prioridade à configuração no arquivo `/etc/apt/preferences`. Seja cuidadoso com essa interferência quando estiver lidando com o arquivo `/etc/apt/preferences`.

3.3 Comandos de sobrevivência do Debian

Com esse conhecimento, é possível viver uma vida de ”atualizações“ **eternas** :-)

3.3.1 Verificar bugs no Debian e procurar ajuda

Se você está tendo problemas sobre um pacote específico, tenha certeza de verificar esses sites antes de pedir ajuda ou antes de enviar um relatório de bug. (`lynx`, `links`, e `w3m` funcionam igualmente bem):

```
$ lynx http://bugs.debian.org/
$ lynx http://bugs.debian.org/nome-do-pacote  # se você sabe o nome do pacote
$ lynx http://bugs.debian.org/número-do-bug   # se você sabe o número do bug
```

Procure no Google (www.google.com) com palavras de busca incluindo "site:debian.org".

Quando tiver dúvidas, leia o manual. Configure `CDPATH` como a seguir:

```
export CDPATH=.:/usr/local:/usr/share/doc
```

e digite

```
$ cd nome-do-pacote
$ pager README.Debian # case esse exista
$ mc
```

3.3.2 Resolvendo problemas com atualização pelo APT

Podem ocorrer problemas de dependência ao atualizar na `unstable/testing`, como descrito em 'Atualizar o sistema Debian' on page 4. Na maior parte das vezes, isso ocorre porque um pacote que será atualizado tem uma nova dependência que é desconhecida. Esses problemas são resolvidos usando

```
# apt-get dist-upgrade
```

Se isso não resolver, então repita um dos seguintes até que o problema se resolva:

```
# apt-get upgrade -f        # continua atualização depois do erro
... ou
# apt-get dist-upgrade -f   # continua atualização distribuição depois do err
```

Alguns scripts de atualização realmente quebrados podem causar um problema persistente. É melhor resolver esse tipo de situação inspecionando os scripts `/var/lib/dpkg/info/nome-do-pacote.{post,pre}{inst,rm}` do pacote problemático e então executar:

```
# dpkg --configure -a    # configura todos os pacotes parcialmente instalados
```

Se um script reclamar de um arquivo de configuração inexistente, procure o arquivo de configuração correspondente em `/etc`. Se existir um com uma extensão `.new` (ou algo similar), modifique-o (`mv`) para remover o sufixo.

Podem ocorrer problemas de dependência ao instalar em `unstable/testing`. Há maneiras de enganar dependências.

```
# apt-get install -f pacote # anula dependências quebradas
```

Um método alternativo para resolver essas situações é usar o pacote `equivs`. Veja `/usr/share/doc/equivs/README.Debian`.

3.3.3 Resgate usando o dpkg

A recuperação de uma falha no `dselect` (APT) pode ser feita em um sistema realmente quebrado usando apenas o `dpkg` sem o APT:

```
# cd /var/cache/apt/archives
# dpkg -i libc6* libdb2* perl*
# dpkg -i apt* dpkg* debconf*
# dpkg -i *  # até que não ocorram erros
```

Se estiver faltando um pacote, pegue-o a partir dos sites espelhos (http://www.debian.org/mirror/) com:

```
# mc            # use "link FTP" apontando para o servidor FTP Debian
```

Recentemente, os pacotes atuais no servidor HTTP/FTP podem não ser localizados sob o clássico diretório `/dist`, mas sob o novo diretório `/pool`.

Então instale com:

```
# dpkg -i /var/cache/apt/archives/arquivodepacote.deb
```

Para uma dependência quebrada, conserte-a ou use:

```
# dpkg --ignore-depends=pacote1,... -i arquivodopacote.deb
# dpkg --force-depends -i arquivodopacote.deb
# dpkg --force-depends --purge pacote
# dpkg --force-confmiss -i arquivodopacote.deb  # Instala arquivo de configur
```

3.3.4 Recuperação dos dados de seleção de pacotes

Se o arquivo `/var/lib/dpkg/status` se corromper por alguma razão, o sistema Debian perde os dados de seleção de pacotes e é severamente prejudicado. Procure pelo antigo arquivo `/var/lib/dpkg/status` em `/var/lib/dpkg/status-old` ou `/var/backups/dpkg.status.*`.

Manter `/var/backups/` em uma partição separada pode ser uma boa idéia já que esse diretório contém muitos dados importantes do sistema.

Se o arquivo `/var/lib/dpkg/status` antigo não estiver disponível, você ainda pode recuperar a informação a partir dos diretórios em `/usr/share/doc/`.

```
# ls /usr/share/doc | \
  grep -v [A-Z] | \
  grep -v '^texmf$' | \
  grep -v '^debian$' | \
  awk '{print $1 " install"}' | \
  dpkg --set-selections
# dselect --expert # reinstalar o sistema, de-selecione conforme necessário
```

3.3.5 Resgate do sistema após quebrar `/var`

Como o diretório `/var` contém dados atualizados regularmente como correio, ele é propício a ser corrompido. Separá-lo em uma partição independente limita os riscos. Se acontecer um desastre, você pode ter que recriar o diretório `/var` para resgatar seu sistema Debian.

Obtenha um conteúdo esqueleto do diretório `/var` a partir de um sistema Debian mínimo funcional baseado na mesma versão Debian ou mais antiga, por exemplo `var.tar.gz` (`http://people.debian.org/~osamu/pub/`), e coloque-o no diretório raiz do sistema quebrado. Então

```
# cd /
# mv var var-old      # se algum conteúdo útil foi deixado
# tar xvzf var.tar.gz # use o arquivo esqueleto do Woody
# aptitude            # ou dselect
```

Isso deve prover um sistema funcional. Você pode realizar a recuperação das seleções de pacotes usando a técnica descrita em 'Recuperação dos dados de seleção de pacotes' on the current page. ([FIXME]: Esse procedimento precisa mais experimentos para verificar.)

3.3.6 Instalar um pacote em um sistema não iniciável

Inicie no Linux usando um disquete/CD de resgate do Debian ou uma partição alternativa em um sistema Linux com multi-inicialização. Monte o sistema não iniciável em `/target` e use o modo de instalação chroot do `dpkg`.

```
# dpkg --root /target -i arquivodopacote.deb
```

Então configure e resolva os problemas.

Por outro lado, se a única coisa que impede de iniciar é um `lilo` quebrado, você pode iniciar usando um disco de resgate padrão do Debian. No prompt de boot, considerando que a partição de sua instalação Linux está em `/dev/hda12` e você quer o nível de execução (runlevel) 3, entre:

```
boot: rescue root=/dev/hda12 3
```

Então você terá iniciado em um sistema quase totalmente funcional com o kernel no disquete. (Podem ocorrer problemas menores devido à falta de recursos do kernel ou módulos.)

3.3.7 O que fazer se o comando `dpkg` estiver quebrado

Um `dpkg` pode tornar impossível instalar qualquer arquivo `.deb`. Um procedimento como o seguinte o ajudará a recuperar-se dessa situação. (Na primeira linha, você pode substituir "links" com o comando de seu browser favorito.)

```
$ links http://http.us.debian.org/debian/pool/main/d/dpkg/
  ... baixe o dpkg_versão_arquit.deb bom
$ su
password: *****
# ar x dpkg_versão_arquitetura.deb
# mv data.tar.gz /data.tar.gz
# cd /
# tar xzfv data.tar.gz
```

Para `i386`, `http://packages.debian.org/dpkg` também pode usado como a URL.

3.4 Comandos de nirvana do Debian

A **Iluminação** com esses comandos salvará uma pessoa da eterna agonia cármica do inferno da atualização e a levará a alcançar o **nirvana** do Debian. :-)

3.4.1 Informação sobre um arquivo

Para encontrar o pacote ao qual um arquivo particular pertence:

```
$ dpkg {-S|--search} padrão # procura pelo padrão nos pacotes instalados
$ wget http://ftp.us.debian.org/debian/dists/sarge/Contents-i386.gz
$ zgrep -e pattern Contents-i386.gz
            # procura o padrão-nome-de-arquivo no arquivo do debian
```

Ou use comandos especializados de pacote:

```
# apt-get install dlocate
                # conflita com slocate (versão segura de locate)
$ dlocate nome-de-arquivo  # alternativa rápida ao dpkg -L e dpkg -S
...
# apt-get install auto-apt # ferramenta de instalação de pacotes sob demanda
# auto-apt update          # cria arquivo db para o auto-apt
$ auto-apt search padrão
               # procura pelo padrão em todos os pacotes, instalados ou não
```

3.4.2 Informação sobre um pacote

Procura e exibe informação a partir dos arquivos de pacotes. Certifique-se de apontar o APT para o(s) repositório(s) próprio(s) editando `/etc/apt/sources.list`. Se você deseja ver como pacotes na `testing`/`unstable` estão em comparação com os atualmente instalados, use `apt-cache policy`—muito legal.

```
# apt-get   check          # atualiza o cache e verifica pacotes quebrados
$ apt-cache search  padrão # procura pacote pela descrição do texto
$ apt-cache policy  pacote # prioridade de pacote/informação de dists
$ apt-cache show -a pacote # exibe descrição do pacote em todas dists
$ apt-cache showsrc pacote # exibe descrição de pacote fonte determinado
$ apt-cache showpkg pacote # informação do pacote para depuração
# dpkg  --audit|-C         # procura pacotes parcialmente instalados
$ dpkg {-s|--status} pacote ... # descrição de pacote instalado
$ dpkg -l pacote ...       # estado de pacote instalado (1 linha cada)
$ dpkg -L pacote ...       # lista nomes de arquivos instalados pelo pacote
```

O `apt-cache showsrc` não foi documentado na versão Woody mas funciona :)

Você também pode encontrar informação sobre pacote em (eu uso o `mc` para navegar entre eles):

```
/var/lib/apt/lists/*
/var/lib/dpkg/available
```

A comparação dos seguintes arquivos dá informação sobre o que exatamente aconteceu nas últimas sessões de instalação.

```
/var/lib/dpkg/status
/var/backups/dpkg.status*
```

3.4.3 Instalação com APT sem acompanhamento

Para uma instalação sem acompanhamento, adicione a seguinte linha em `/etc/apt/apt.conf`:

```
Dpkg::Options {"--force-confold";}
```

Isso é equivalente a executar `apt-get -q -y` *`nome-do-pacote`*. Como isso responde "yes" automaticamente a todas as perguntas, pode causar problemas, então use essa dica com cuidado. Veja `apt.conf(5)` e `dpkg(1)`.

Você pode configurar quaisquer pacotes particulares mais tarde seguindo 'Reconfigurar pacotes instalados' on this page.

3.4.4 Reconfigurar pacotes instalados

Use o seguinte para reconfigurar qualquer pacote já instalado.

```
# dpkg-reconfigure --priority=medium pacote [...]
# dpkg-reconfigure --all   # reconfigura todos os pacotes
# dpkg-reconfigure locales # gera qualquer locale extra
# dpkg-reconfigure --p=low xserver-xfree86 # reconfigura servidor X
```

Faça isso para o `debconf` se você quiser mudar o modo de diálogo do `debconf` permanentemente.

Alguns programas vêm com scripts de configuração especiais.

```
apt-setup     - cria /etc/apt/sources.list
install-mbr   - instala um gerenciador de Master Boot Record
tzconfig      - configura o fuso horário local
gpmconfig     - configura o daemon de mouse gpm
sambaconfig   - configura o Samba no Potato (Woody usa o debconf)
eximconfig    - configura o Exim (MTA)
texconfig     - configura o teTeX
apacheconfig  - configura o Apache (httpd)
cvsconfig     - configura o CVS
sndconfig     - configura o sistema de som
...
update-alternatives - define o comando padrão, por exemplo, vim como vi
update-rc.d         - gerenciamento de script de init System-V
update-menus        - sistema de menus Debian
...
```

3.4.5 Remover e eliminar (purge) pacotes

Remover um pacote mantendo sua configuração:

```
# apt-get remove pacote ...
# dpkg  --remove pacote ...
```

Remover um pacote e toda sua configuração:

```
# apt-get remove --purge pacote ...
# dpkg    --purge        pacote ...
```

3.4.6 Mantendo pacotes antigos

Por exemplo, o travamento de `libc6` e `libc6-dev` para o `dselect` e `apt-get -u upgrade` *`pacote`* pode ser feito como a seguir:

```
# echo -e "libc6 hold\nlibc6-dev hold" | dpkg --set-selections
```

`apt-get -u install` *`pacote`* não será impedido por este "hold". Para manter um pacote forçando um rebaixamento automático para `apt-get -u upgrade` *`pacote`* ou `apt-get -u dist-upgrade`, adicione o seguinte a `/etc/apt/preferences`:

```
Package: libc6
Pin: release a=stable
Pin-Priority: 2000
```

Aqui o campo "`Package:`" não pode usar entradas como "`libc6*`". Se você precisa manter todos os pacotes binários relacionados ao pacote fonte `glibc` em uma versão sincronizada, precisa listá-los explicitamente.

O seguinte listará pacotes em "hold":

```
dpkg --get-selections "*"|grep -e "hold$"
```

3.4.7 Sistema híbrido `stable`/`testing`/`unstable`

`apt-show-versions` pode listar versões de pacotes disponíveis por distribuição.

```
$ apt-show-versions | fgrep /testing | wc
... quantos pacotes você tem da testing
$ apt-show-versions -u
... lista de pacotes atualizáveis
$ apt-get install ‘apt-show-versions -u -b | fgrep /unstable‘
... atualiza todos os pacotes unstable para suas versões mais novas
```

3.4.8 Descartar arquivos de pacote em cache

A instalação de pacotes com o APT deixa pacotes em cache em `/var/cache/apt/archives` e esses precisam ser excluídos.

```
# apt-get autoclean # remove apenas os arquivos de pacotes sem uso
# apt-get clean     # remove todos os arquivos de pacotes em cache
```

3.4.9 Gravar/copiar a configuração do sistema

Para fazer uma cópia local dos estados de seleção dos pacotes:

```
$ dpkg --get-selections "*" >minhas_seleções   # ou use \*
```

`"*"` faz *`minhas_seleções`* incluir entradas de pacotes para "purge" também.

Você pode transferir esse arquivo para outro computador, e instalá-lo lá com:

```
# dselect update
# dpkg --set-selections <minhas_seleções
# apt-get -u dselect-upgrade    # ou dselect install
```

3.4.10 Portar um pacote para o sistema `stable`

Para atualizações parciais do sistema `stable`, reconstruir um pacote dentro de seu ambiente usando o pacote fonte é desejável. Isso evita atualizações muito grandes de pacotes devido a suas dependências. Primeiro, adicione as seguintes entradas ao `/etc/apt/sources.list`:

```
deb-src http://http.us.debian.org/debian testing \
 main contrib non-free
deb-src http://non-us.debian.org/debian-non-US testing/non-US \
 main contrib non-free
deb-src http://http.us.debian.org/debian unstable \
 main contrib non-free
deb-src http://non-us.debian.org/debian-non-US unstable/non-US \
 main contrib non-free
```

Aqui cada entrada para `deb-src` está quebrada em duas linhas devido a restrições de impressão, mas a entrada real em `sources.list` deve consistir de uma única linha.

Então pegue a fonte e faça um pacote local:

```
$ apt-get update  # atualiza a lista de busca de pacotes fontes
$ apt-get source pacote
$ dpkg-source -x pacote.dsc
$ cd pacote-versão
  ... inspecione pacotes necessários (Build-depends no arquivo .dsc) e
      instale-os também.  Você também precisa do pacote "fakeroot".

$ dpkg-buildpackage -rfakeroot

  ...ou (sem assinatura)
$ dpkg-buildpackage -rfakeroot -us -uc # use "debsign" depois se necessário

  ...Então para instalar
$ su -c "dpkg -i arquivodopacote.deb"
```

Geralmente, é necessário instalar uns poucos pacotes com o sufixo "-dev" para satisfazer dependências de pacotes. O `debsign` está no pacote `devscripts`. O `auto-apt` pode facilitar a satisfação dessas dependências. O uso de `fakeroot` evita o uso desnecessário da conta root.

No Woody, esses problemas de dependências podem ser simplificados. Por exemplo, para compilar um pacote `pine` de apenas fonte:

```
# apt-get build-dep pine
# apt-get source -b pine
```

3.4.11 Repositório local de pacotes

Para criar um repositório local de pacotes que seja compatível com o APT e o sistema `dselect`, é necessário criar o arquivo `Packages` e os arquivos de pacotes precisam ser colocados em uma árvore de diretórios específica.

Pode-se fazer um repositório `deb` local similar a um repositório Debian oficial assim:

```
# apt-get install dpkg-dev
# cd /usr/local
# install -d pool # os pacotes físicos são localizados aqui
# install -d dists/unstable/main/binary-i386
# ls -1 pool | sed 's/_.*$/ prioridade/ seção/' | uniq > override
# editor override # ajuste prioridade e seção
# dpkg-scanpackages pool override /usr/local/ \
   > dists/unstable/main/binary-i386/Packages
# cat > dists/unstable/main/Release << EOF
Archive: unstable
Version: 3.0
Component: main
```

```
Origin: Local
Label: Local
Architecture: i386
EOF
# echo "deb file:/usr/local unstable main" \
   >> /etc/apt/sources.list
```

Alternativamente, pode-se fazer um repositório `deb` local curto e grosso:

```
# apt-get install dpkg-dev
# mkdir /usr/local/debian
# mv /algum/lugar/pacote.deb /usr/local/debian
# dpkg-scanpackages /usr/local/debian /dev/null | \
  gzip - > /usr/local/debian/Packages.gz
#  echo "deb file:/usr/local/debian ./" >> /etc/apt/sources.list
```

Esses repositórios podem ser acessados remotamente disponibilizando acesso a esses diretórios através de métodos HTTP ou FTP e mudando as entradas em `/etc/apt/sources.list` apropriadamente.

3.4.12 Converter ou instalar um pacote binário alien

O `alien` possibilita a conversão de pacotes binários providos em formatos de arquivo `rpm` da Redhat, `slp` da Stampede, `tgz` do Slackware, e `pkg` do Solaris em um pacote `deb` do Debian. Se você quiser usar um pacote de outra distribuição Linux ao invés do que tem instalado em seu sistema, você pode usar o `alien` para convertê-lo para seu formato de pacotes preferido e instalá-lo. O `alien` também suporta pacotes LSB.

3.4.13 Verificar arquivos de pacotes instalados

`debsums` possibilita a verificação de arquivos de pacotes instalados com checksums MD5. Alguns pacotes não têm checksums MD5 disponíveis. Uma solução temporária para administradores de sistema:

```
# cat >>/etc/apt/apt.conf.d/90debsums
DPkg::Post-Install-Pkgs {"xargs /usr/bin/debsums -sg";};
^D
```

por Joerg Wendland <joergland@debian.org> (não testado).

3.5 Outras peculiaridades do Debian

3.5.1 O comando `dpkg-divert`

Desvios (divertions) de arquivos são uma maneira de forçar o `dpkg` a não instalar um arquivo em sua localização padrão, mas em uma localização **desviada**. Os **desvios** podem ser usados através dos scripts de pacotes Debian para desviar de um arquivo quando ele causa um conflito. Os administradores de sistema também podem usar um desvio para substituir o arquivo de configuração de um pacote, ou quando alguns arquivos (que não estão marcados como **conffiles**) precisam ser preservados pelo `dpkg`, ao instalar uma versão mais nova de um pacote que contém aqueles arquivos .

```
# dpkg-divert [--add]  nomedoarquivo # adiciona "desvio"
# dpkg-divert --remove nomedoarquivo # remove "desvio"
```

Geralmente é uma boa idéia não usar o `dpkg-divert` a menos que ele seja absolutamente necessário.

3.5.2 O pacote `equivs`

Se você compilar um programa a partir da fonte, é melhor torná-lo um pacote local debianizado real (`*.deb`). Use `equivs` como último recurso.

```
Package: equivs
Priority: extra
Section: admin
Description: Engana dependências de pacote Debian
 Este é um pacote falso que pode ser usado para criar pacotes
 Debian, que contém apenas informação de dependência.
```

3.5.3 Comandos alternativos

Para fazer o comando `vi` executar `vim`, use `update-alternatives`:

```
# update-alternatives --display vi
...
# update-alternatives --config vi
  Selection    Command
-----------------------------------------------
      1        /usr/bin/elvis-tiny
      2        /usr/bin/vim
*+    3        /usr/bin/nvi

Enter to keep the default[*], or type selection number: 2
```

Os itens no sistema de alternativas do Debian são mantidos em `/etc/alternatives` como ligações simbólicas.

Para definir seu ambiente X Window favorito, aplique `update-alternatives` para `/usr /bin/x-session-manager` e `/usr/bin/x-window-manager`.

`/bin/sh` é uma ligação simbólica direta para `/bin/bash` ou `/bin/dash`. É mais seguro usar `/bin/bash` para ser compatível com scripts contaminados com velho Bashismo, mas é mais disciplinado usar `/bin/dash` para reforçar a compatibilidade POSIX. A atualização para um kernel Linux 2.4 tende a definir isso para `/bin/dash`.

3.5.4 `init` System-V e níveis de execução (runlevels)

O nível de execução padrão em que inicializar pode ser definido em `/etc/inittab`.

Diferente de outras distribuições, o Debian deixa o gerenciamento de nível de execução completamente como responsabilidade do administrador do sistema. O gerenciamento de `init` estilo System-V no Debian deve ser feito através de scripts `update-rc.d`.

Iniciar `/etc/init.d/`*`nome`* no nível de execução 1,2,3 e parar em 4,5 com número de prioridade de seqüência 20 (normal) pode ser feito com:

```
# update-rc.d nome start 20 1 2 3 . stop 20 4 5 .
```

A remoção de ligações simbólicas enquanto o script em `init.d` ainda existe pode ser feita com:

```
# update-rc.d -f nome remove
```

Para editar níveis de execução, eu trapaceio. Eu edito as entradas manualmente usando o comando `mv` no prompt shell do `mc` ao copiar as entradas de ligação usando Alt-Enter. Por exemplo:

```
# mv S99xdm K99xdm # desabilita xdm (X display manager)
```

Eu também desabilito o daemon inserindo `exit 0` no início de um script `init.d` como um hack rápido. Esses são os arquivos de configuração depois de tudo.

3.5.5 Serviços daemon desabilitados

A distribuição Debian considera a segurança do sistema seriamente e espera que o administrador do sistema seja competente. Assim, algumas vezes a facilidade de uso aparece como assunto secundário e muitos serviços daemon vêm com o maior nível de segurança, com os mínimos serviços (ou nenhum) disponíveis como seu estado de instalação padrão.

Execute `ps aux` ou verifique os conteúdos de `/etc/init.d/*` e `/etc/inetd.conf`, se tiver qualquer dúvida (sobre Exim, DHCP, . . .). Verifique também `/etc/hosts.deny`. O comando `pidof` também é útil (veja `pidof(8)`).

O X11 não permite conexões TCP/IP (remotas) por padrão nas versões recentes do Debian. O redirecionamento do X em SSH também é desabilitado.

Apêndice A

Apêndice

A.1 Autores

O Referência Rápida Debian foi iniciado por Osamu Aoki <osamu@debian.org> como um registro pessoal de instalação que foi eventualmente chamado "Referência Rápida". Muitas partes foram extraídas dos arquivos da lista de discussão "debian-user". Além disso, também foram usados o "Manual de Instalação Debian" e "Notas de Lançamento Debian"

Seguindo uma sugestão de Josip Rodin, que é muito ativo no Projeto de Documentação Debian (http://www.debian.org/doc/ddp) (DDP) e é o atual mantenedor da "Debian FAQ", esse documento foi renomeado como "Referência Debian" e foi fundido com vários capítulos da "Debian FAQ" com conteúdo de referência. Então a "Referência Rápida Debian" é formada de resumos.

Esse documento foi editado, traduzido, e ampliado pelos seguintes membros do time QREF:

- Originais em inglês do "Referência Rápida..." original
  - Osamu Aoki <osamu@debian.org> (líder: todos os conteúdos)
- Revisão do inglês e contribuição adicional
  - David Sewell <dsewell@virginia.edu> (líder: estilo en)
  - Thomas Hood <jdthood@yahoo.co.uk> (relacionado a rede)
  - Brian Nelson <nelson@bignachos.com> (especialmente relacionado ao X)
  - Jan Michael C Alonzo <jmalonzo@spaceants.net>
  - Daniel Webb <webb@robust.colorado.edu>
  - Retorno de todos os tradutores
- Tradução para o francês
  - Guillaume Erbs <gerbs@free.fr> (líder: fr)
  - Rénald Casagraude <rcasagraude@interfaces.fr>
  - Jean-Pierre Delange <delange@imaginet.fr>
  - Daniel Desages <daniel@desages.com>
- Tradução para o italiano
  - Davide Di Lazzaro <mc0315@mclink.it> (líder: it)
- Tradução para o português (Brasil)
  - Paulo Rogério Ormenese <pormenese@uol.com.br> (líder: pt-br)

  - Andre Luis Lopes <andrelop@ig.com.br>
  - Marcio Roberto Teixeira <marciotex@pop.com.br>
  - Rildo Taveira de Oliveira <to_rei@yahoo.com>
  - Raphael Bittencourt Simoes Costa <raphael-bsc@bol.com.br>
  - Gustavo Noronha Silva <kov@debian.org> (coordinator)
- Tradução para o espanhol
  - Walter Echarri <wecharri@infovia.com.ar> (líder: es)
  - José Carreiro <ffx@urbanet.ch>
- Tradução para o alemão
  - Jens Seidel <tux-master@web.de> (líder: de)
  - Willi Dyck <wdyck@gmx.net>
  - Stefan Schröder <stefan@fkp.uni-hannover.de>
  - Agon S. Buchholz <asb@kefk.net>
- Tradução para o polonês—os seguintes membros do PDDP (http://debian.linux.org.pl):
  - Marcin Andruszkiewicz
  - Mariusz Centka <mariusz.centka@debian.linux.org.pl>
  - Bartosz Fenski <fenio@debian.linux.org.pl> (líder: pl)
  - Radoslaw Grzanka <radekg@debian.linux.org.pl>
  - Bartosz 'Xebord' Janowski
  - Jacek Lachowicz
  - Rafal Michaluk
  - Leonard Milcin, Jr.
  - Tomasz Z. Napierala <zen@debian.linux.org.pl>
  - Oskar Ostafin <cx@debian.linux.org.pl>
  - Tomasz Piekos
  - Jacek Politowski
  - Mateusz Prichacz <mateusz@debian.linux.org.pl>
  - Marcin Rogowski
  - Pawel Rozanski
  - Mariusz Strzelecki
  - Krzysztof Scierski
  - Przemyslaw Adam Smiejek <tristan@debian.linux.org.pl>
  - Mateusz Tryka <uszek@debian.linux.org.pl>
  - Cezary Uchto
  - Krzysztof Witkowski <tjup@debian.linux.org.pl>
  - Bartosz Zapalowski <zapal@debian.linux.org.pl>
- Tradução para o chinês (simplificado)
  - Hao "Lyoo" LIU <iamlyoo@163.net> (líder: zh-cn)
  - Ming Hua <minghua@rice.edu>
- Tradução para o chinês (tradicional)
  - Asho Yeh <asho@debian.org.tw> (líder: zh-tw)
  - Tang Wei Ching <wctang@csie.nctu.edu.tw> (ex-líder: zh-tw)
- Tradução para o japonês
  - Shinichi Tsunoda <tsuno@ngy.1st.ne.jp> (líder: ja)
  - Osamu Aoki <osamu@debian.org>

A.2 Garantias

Como eu não sou um expert, não pretendo ser totalmente compreensível quanto ao Debian ou ao Linux em geral. As considerações de segurança que faço devem ser aplicáveis apenas ao uso doméstico.

Este documento não substitui nenhum guia oficial.

Não se oferecem garantias de nenhum tipo. Todas as marcas registradas são de propriedade de seus respectivos donos.

A.3 Retorno

Comentários e adendos a esse documento são sempre bem-vindos. Por favor, envie uma mensagem para o sistema BTS Debian (`http://bugs.debian.org/`) relacionada ao pacote `debian-reference` ou para seus respectivos pacotes de traduções. O uso do `reportbug` facilita o envio de um relatório de bug completo. Você pode ainda enviar uma mensagem em inglês para Osamu Aoki (`http://people.debian.org/~osamu/`) em `<osamu@debian.org>` ou para cada tradutor em seus respectivos idiomas.